Review of Literature

2. REVIEW OF LITERATURE

Image classification is a complex process that may be affected by many factors. This chapter examines current practices, problems and prospects of image classification. Section 2.1 provides a review of various proposals made with Tamil character recognition system using general and classification approaches, followed by a summary of studies on general classification techniques used in Section 2.2. A brief description of multi-learner system with emphasis on multi-class classification techniques is provided in Section 2.3 and 2.4. The various classification techniques (both single and multiple) are summarized in Section 2.5. Evaluation of image classification is an important and vital task and is dealt in Section 2.6.

2.1. TAMIL CHARACTER RECOGNITION SYSTEM

During the past thirty years, substantial research efforts have been devoted to character recognition that is used to translate human readable characters to machine-readable codes (Pal and Chaudhuri, 2004). The literature has several studies for English language (Bozinovic and Srihari, 2002; Hu *et al.*, 2006) Chinese/Japanese languages (Deng *et al.*, 2004; Chang, 2006; Yamada *et al.*, 1990) and handwritten numerals (Lee, 1996; Cai and Liu, 1999). However, less attention had been given to Indian language recognition.

Main reasons for this slow development could be attributed to the complexity of the shape of Indian scripts and also the large set of different patterns that exist in these languages, as opposed to English (Kannan and Prabhakar, 2009). Indian scripts are different from Roman script in several ways. Indian scripts are two-dimensional compositions of symbols: core characters in the middle strip, optional modifiers above and/or below core characters. Two characters may be in shadow of each other. While line segments (strokes) are the

predominant features for English, most of the Indian language scripts are formed by curves, holes, and also strokes. In Indian language scripts, the concept of upper case and lower-case characters is absent; however, the alphabet itself contains more number of symbols than that of English. Some efforts have been reported in the literature for Telugu, Devanagari (Bansal and Sinha, 2009; Bajaj and Chaudhary, 2005; Palit and Chaudhary, 2007) and Bangla (Chaudhuri and Pal, 2007) scripts. This section presents the various Tamil OCR studies.

Combining Self-Organizing Maps and Radial Basis Function Networks for Tamil Handwritten Character Recognition was proposed by (Santhosh Baboo, Subashini and Krishnaveni, 2009). An exceptional effort has been extended in making a computer recognize both typed and handwritten characters automatically. Methods currently widely used for character recognition for these languages are mainly those which involve pattern matching using image processing techniques. They experimented with two different approaches. One is SOM based method wherein the interactions between the features in the classification are done using unsupervised learning. In the second approach, a combination of RBF and SOM has been taken to investigate its dynamic training principles in our classification network. The classification ability of RBF-SOM is compared to SOM Network. The comparison is based on the scanned database containing features extracted from preprocessing techniques. The assessment is in terms of average recognition accuracy and the number of training samples required in obtaining an acceptable performance. It also performs error analysis to determine the advisability of combining the classifiers.

This section presents the proposals under Tamil OCR in two sections. The first section discusses the general methods proposed, while the second section presents proposed that use classifiers during OCR.

2.1.1. Studies on OCR

Siromoney et al. (1978) first described a method for recognition of machine printed Tamil characters using an encoded character string dictionary. The scheme employs string features extracted by row- and column-wise scanning of character matrix. The features in each row (column) are encoded suitably depending upon the complexity of the script to be recognized. A given text is presented symbol by symbol and information from each symbol is extracted in the form of a string and compared with the strings in the dictionary. When there is agreement the letters are recognized and printed out in Roman letters following a special method of transliteration. The lengthening of vowels and hardening of consonants are indicated by numerals printed above each letter.

Chinnuswamy et al. (2003) proposed an approach for hand-printed Tamil character recognition. Here, the characters are assumed to be composed of line-like elements, called primitives, satisfying certain relational constraints. Labeled graphs are used to describe the structural composition of characters in terms of the primitives and the relational constraints satisfied by them. The recognition procedure consists of converting the input image into a labeled graph representing the input character and computing correlation coefficients with the labeled graphs stored for a set of basic symbols. This algorithm uses topological matching procedure to compute the correlation coefficients and then maximizes the correlation coefficient.

Suresh et al. (2000) describes an approach to use the fuzzy concept on handwritten Tamil characters to classify them as one among the prototype characters using a feature called distance from the frame and a suitable membership function. The unknown and prototype characters are preprocessed and considered for recognition. The theory of fuzzy set provides an approximate but effective means of describing the behavior of ill-defined systems. Patterns of

human origin like handwritten characters are to some extent found to be fuzzy in nature. It is decided to use fuzzy conceptual approach effectively. The algorithm is tested for about 250 samples for numerals and seven chosen Tamil characters and the success rate obtained varies from 76% to 94%.

Aparna *et al.* (2002) presented a complete document image analysis system for Tamil newsprint. The system includes the full suite of processes from skew correction, binarization, segmentation; text and non-text block classification, line, word and character segmentation and character recognition to final reconstruction. Experience with OCR problems teaches that for most subtasks (text block identification, and character recognition.) involved in OCR, there is no single technique that gives perfect results for every type of document image. We have used the strength of Artificial Neural Networks in empirical model building for solving the key problems of segmentation and character recognition. The final document is reconstructed in HTML document format and it is currently done manually. An important lacuna in our present system is absence of a suitable document model. A document is treated as a collection of disparate items without any logical structure connecting all of them. Future efforts will be focused on embedding various document components into a logical structure. This will help to make the reconstruction process as automatized. 94% recognition rate is obtained, when the text block having a few touching characters present.

Joshi *et al.* (2004) gives a comparison of elastic matching schemes for writer dependent on-line handwriting recognition of isolated Tamil characters. Three different features are considered namely, preprocessed x-y co-ordinates, quantized slope values, and dominant point co-ordinates. Seven schemes based on these three features are compared using an elastic distance measure. The comparison is carried out in terms of recognition accuracy, recognition speed, number of training templates, and dynamic time warping based distance measure has been presented. The results show that dominant points based two-stage

scheme, and combination of rigid and elastic matching schemes perform better than rest of the schemes, especially from the point of view of implementing them in a real time application. Efforts are underway to devise character grouping schemes for hierarchical classification, and classifier combination schemes so as to obtain a computationally more efficient recognition scheme with improved accuracy.

Various strategies and techniques involved in the recognition of Tamil text were studied by Seethalakshmi *et al.* (2005). They refer Optical Character Recognition (OCR) for the process of converting printed Tamil text documents into software translated Unicode Tamil Text. The printed documents available in the form of books, papers, magazines, etc. are scanned using standard scanners which produce an image of the scanned document. As part of the preprocessing phase the image file is checked for skewing. The skewed image is corrected by a simple rotation technique in the appropriate direction, and then it is passed through a noise elimination phase and is binarized. The preprocessed image is segmented using an algorithm, which decomposes the scanned text into paragraphs using special space detection technique and then the paragraphs into lines using vertical histograms, and lines into words using horizontal histograms, and words into character image glyphs using horizontal histograms. Each image glyph is comprised of 32×32 pixels. Thus a database of character image glyphs is created out of the segmentation phase. Then all the image glyphs are considered for recognition using Unicode mapping. Each image glyph is passed through various routines, which extract the features of the glyph. The various features that are considered for classification are the character height, character width, the number of horizontal lines (long and short), the number of vertical lines (long and short), the horizontally oriented curves, the vertically oriented curves, and the number of circles, number of slope lines, image centroid and special dots. The extracted features are passed to a Support Vector Machine (SVM) where the characters are

classified by Supervised Learning Algorithm. These classes are mapped onto Unicode for recognition.

The problem of high interclass similarity in the case of Tamil characters is addressed by Sundaresan and Keerthi (1999) using features like angle features, Fourier coefficients and wavelet features. The results are compared using a Neural Network classifier. In the absence of smoothing, angle features are susceptible to noise and may fail to capture the intra-class similarity. Fourier coefficients do not capture subtle differences between two similar-looking characters because a change in the values of x and y over a small interval of time gets nullified over the entire frequency domain. On the other hand, Wavelet features are shown to retain the intra-class similarity and inter-class differences, resulting in high recognition accuracy. A Single hidden layer network was used for classification. The network gave excellent performance. The classification accuracy is 96.54% for 12-character problem and 94.30% for 135-character problem.

A Data-driven HMM-based online handwritten word recognition system for Tamil language was proposed by (Bharath and Madhvanath, 2007). A symbol set consisting of 84 symbols was defined for the word recognition task and each symbol was modeled using a left-to-right HMM. Intersymbol pen-up strokes were modeled explicitly using two state left-to-right HMMs to capture the relative positions between symbols in the word context. Independently built symbol models and inter-symbol pen-up stroke models were concatenated to form the word models. The relatively low performance in the case of high lexicon size can be improved by the use of statistical language models, which are commonly applied in Western cursive recognition.

A Suitable Framework for Recognition of Epigraphical Tamil Scripts using Unsupervised Learning Algorithm was proposed by (P.Subashini, M.Krishnaveni, N.Sridevi, 2011).Tamil, the south Indian language, is the oldest of all in the world.

The need for OCR arises in the context of digitizing Tamil documents from the ancient and old era to the latest, which helps in sharing the data. Most of the work carried so far uses traditional methodologies which are not effective. An inclusive study and experiments is carried out here to find suitable methods to construct a better recognition system. This system uses the unsupervised learning algorithm for classification and recognition. The system is experimented with two different approaches. One is SOM based method wherein the interactions between the features are well supported and the other is BPN. The performance of the system was measured using sum squared error function and the overall performance of the system is 95%.

2.1.2. Classification-Based Approaches

Hewavitharana and Fernando (2002) a system is described to recognize handwritten Tamil characters using a two-stage classification approach, for a subset of the Tamil alphabet, which is a hybrid of structural and statistical techniques. In the first stage, an unknown character is pre-classified into one of the three groups: core, ascending and descending characters. Structural properties of the text line are used for this classification. Then, in the second stage, members of the pre-classified group are further analyzed using a statistical classifier for final recognition. The main recognition errors were due to abnormal writing and ambiguity among similar shaped characters. This could be avoided by using a word dictionary to look-up for possible character compositions. The presence of contextual knowledge will help to eliminate the ambiguity. They strongly feel that the method of pre-classification would have much higher recognition accuracy if applied to Optical Character Recognition, since printed characters preserve the correct positioning on three-zone frame.

Subashini, Krishnaveni and Sridevi (2011) described a Period Prediction System for Tamil Epigraphical Scripts based on Support Vector Machine.

Epigraphical scripts are the inscription written on various materials and the study of it is vital in knowing the civilized past and hence classification of character belonging to various periods is imperative before using the character bank of the particular period. Therefore a system is proposed for prediction of the period and it is being done by examining a few character referred to as test characters in Tamil language. These test characters are sampled from the script automatically and matched with the characters available for different periods using machine intelligence. The proposed system here has various modules like binarization, thinning, segmentation, feature extraction and finally classification and period prediction using Support Vector Machine. The performance of the system is measured using the four parameters such as prediction rate, Correction rate, Error rate and Time taken to predict the centuries. The system achieves overall accuracy of 90.45%.

Shanthi and Duraiswamy (2007) described a recognition system for offline unconstrained handwritten Tamil characters based on Support Vector Machine (SVM). SVM is a new type of pattern classifier based on a novel statistical learning technique. Due to the difficulty in great variation among handwritten characters, the system is trained with 106 characters and tested for 34 selected Tamil characters. The characters are chosen such that the sample data set represents almost all the characters. Data samples are collected from different writers on A4 sized documents. They are scanned using a flat bed scanner at a resolution of 300 dpi and stored as grey scale images. Various preprocessing operations are performed on the digitized image to enhance the quality of the image. Random sized preprocessed image is normalized to uniform sized image. Pixel densities are calculated for different zones of the image and these values are used as the features of a character. These features are used to train and test the support vector machine.

Shivsubramani et al. (2007) presents an efficient method for recognizing printed Tamil characters exploring the interclass relationship between them, which should be accomplished using Multiclass Hierarchical Support Vector Machines. A new variant of Multi Class Support Vector Machine constructs a hyper plane that separates each class of data from other classes. Character recognition, thus, involves classification of characters into multi-classes. Of the 126 unique characters identified in Tamil language, inter-class dependencies were found within many characters due to the similarity in their shapes. This enabled them to be organized into hierarchies, thus enhancing and simplifying the process classification.

Taking advantage of the inter-class dependencies within the character a hierarchical based classification based on these views was put forth by Szedmak et al. (2005). Combining both the views together, a Multiclass Hierarchical SVM algorithm was devised and is understood to be very efficient methods for character classification. The algorithm did prove more efficient than some of the commonly used classifiers. Some merits of our algorithm are:

- Strong mathematical model foundation rather than heuristics and analogies.
- Efficient in terms of accuracy (96.85%) in comparison with many commonly used classifiers.

A generalized framework for Indic script character recognition and Tamil character recognition was proposed by Aparna *et al.*, (2004). Unique strokes in the script are manually identified and each stroke is represented as a string of shape features. The test stroke is compared with the database of such strings using the proposed flexible string-matching algorithm. The sequence of stroke labels is then converted into "horizontal block" using a rule list and the sequence of horizontal blocks is recognized as a character (with its IISCI code) using a Finite State

Automaton (FSA). Online HWR studies typically handle smaller number of stroke classes since many of them deal with Latin script (26 or 52 classes).

Deepu and Madhvanath (2004) and Joshi *et al.* (2004) proposed a subspace-based method using Principal Component Analysis (PCA) for Tamil character recognition. The input is a temporally ordered sequence of (x, y) pen coordinates corresponding to an isolated character obtained from a digitizer. The input is converted into a feature vector of constant dimensions following smoothing and normalization. Each class is modeled as a subspace, and for classification, the orthogonal distance of the test sample to the subspace of each class is computed. The effort published in [30] compares the performance of DTW and PCA for three modes of recognition: writer independent, writer dependent and writer adaptive. DTW is shown to outperform PCA in all the three modes of recognition. Although the performance of DTW based method is marginally better, in terms of speed, subspace based method wins over. Also classifier combination schemes for the two methods are proposed. DTW based method is computationally expensive; the disadvantage may be overcome by using prototype selection/reduction methods.

Sutha and Ramaraj (2007) describe an approach to recognize handwritten Tamil characters using a multilayer perception with one hidden layer. The feature extracted from the handwritten character is Fourier descriptors. Also an analysis was carried out to determine the number of hidden layer nodes to achieve high performance of back propagation network in the recognition of handwritten Tamil characters. The system was trained using several different forms of handwriting provided by both male and female participants of different age groups. Test results indicate that Fourier descriptors combined with back propagation network provide good recognition accuracy of 97% for handwritten Tamil characters.

There are few works going on in Tamil OCR, the accuracy of the approaches still remain a challenging area of research. From the review study, it

can be seen that accurate classification-based character recognition is still elusive and this research work focus on this areas.

2.2. GENERAL CLASSIFICATION TECHNIQUES

Classification is an important problem and many classification models have been proposed in the literature. Weiss and Kulikowski (1991) and Michie *et al.* (2004) provide overviews of classification methods. While classification is a well-studied problem, only recently has there been focus on algorithms that can handle large databases. The intuition is that by classifying larger datasets, the accuracy of the classification model can be improved (Catlett, 1991; Chan and Stolfo (1993a, 1993b). Several classification models have been proposed over the years, e.g. Neural Networks (Lippmann, 1987), statistical models like linear/quadratic discriminants (James, 1985), decision trees (Breiman *et al.*, 2007) and genetic models (Goldberg, 2006). Among these models, until recently, decision trees were considered particularly suited for data mining (Agrawal *et al.*, 1993; Mehta *et al.*, 1996). Decision trees can be constructed relatively fast compared to other methods and are simple and easy to understand. Moreover, trees can be easily converted into SQL statements that can be used to access databases efficiently (Agrawal *et al.*, 1992). Finally, decision tree classifiers obtain similar and sometimes better accuracy when compared with other classification methods (Michie *et al.*, 2004).

This section reviews research work based on classification algorithms. Most algorithms in the machine learning and statistics community are main memory algorithms, even though today's databases are in general much larger than main memory. There have been several approaches to dealing with large databases. One approach is to discretize each ordered attribute and run the algorithm on the discretized data.

All discretization methods for classification that take the class label into account when discretizing assume that the database fits into main memory

(Quinlan, 1983; Fayyad and Irani, 1993; Maass, 1994; Dougherty *et al.*, 1995). Catlett (1991) proposed sampling at each node of the classification tree, but considers in his studies only datasets that could fit in main memory. Methods for partitioning the dataset such that each subset fits in main memory are considered by Chan and Stolfo (1993a, 199b); although this method enables classification of large datasets their studies show that the quality of the resulting decision tree is worse than that of a classifier that was constructed taking the complete database into account at once.

Agrawal *et al.* (1992) introduce an interval classifier that could use database indices to efficiently retrieve portions of the classified dataset using SQL queries. However, the method does not scale to large training sets (Shafer *et al.*, 1996).

Fukuda *et al.* (1996) construct decision trees with two dimensional splitting criteria. Although their algorithm can produce rules with very high classification accuracy, scalability was not one of the design goals. In addition, the decision tree no longer has the intuitive representation of a tree with one-dimensional splits at each node.

The decision tree classifier in Mehta (1996), called SLIQ, was designed for large databases but uses an in-memory data structure that grows linearly with the number of tuples in the training database. This limiting data structure was eliminated in Shafer *et al.* (1996), which introduced Sprint, a scalable classifier.

In another work, Morimoto *et al.* (1988) developed algorithms for categorical predictor variables with large domains; the emphasis of this work is to improve the quality of the resulting tree. Rastogi and Shim (1998) developed PUBLIC, a scalable decision tree classifier using top-down pruning. Since pruning is an orthogonal dimension to tree growth, their techniques can be easily incorporated into our schema.

2.3. MULTI-LEARNER SYSTEMS

While a variety of multiple classifier systems have been studied since at least the late 1950's, this area came alive in the 90's with significant theoretical advances as well as numerous successful practical applications. Multiple classifier systems are special cases of approaches that integrate several data-driven models for the same problem. A key goal is to obtain a better composite global model, with more accurate and reliable estimates or decisions. In addition, modular approaches often decompose a complex problem into sub problems for which the solutions obtained are simpler to understand, as well as to implement, manage and update.

Multilearner systems have a rather long and interesting history. For example, Borda counts for combining multiple rankings are named after its 18th century French inventor, Jean-Charles de Borda. Early notable systems include Selfridge's Pandemonium (Selfridge, 1958), a model of human information processing involving multiple demons. Each demon was specialized for detecting specific features or classes. A head-demon (the combiner) would select the best demon. This scheme is referred to as a "winner-take-all" solution now-a-days.

Nilsson's committee machine (Nilsson, 1965) combined several linear two-class models to solve a multiclass problem. A strong motivation for multilearner systems was voiced by Kanal (1974). This inspired much work in the late seventies on combining linguistic and statistical models and on combining heuristic search with statistical pattern recognition. Subsequently, similar sentiments on the importance of multiple approaches were also voiced in the AI community, e.g., by Minsky (1991),

In the 80's, integration of multiple data sources and/or learned models was considered in several disciplines, for example, the combining of estimators in econometrics (Granger, 1989) and evidences in rule-based systems. Especially

noteworthy are consensus theoretic methods developed in statistics and management science, including how to produce a single probability distribution that summarizes multiple estimates from different Bayesian experts (French, 1985; Benediktsson and Swain, 1992).

The area of decision fusion and multi-sensor data fusion has a rich literature from this era that can be useful for modern day multi_classifier problems as well (Luo and Kay, 1989). Multiple model systems are also encountered in some large engineering systems such as those that demand fault tolerance or employing control mechanisms that may need to function in different operating regimes (Narendra *et al.*, 1995). In particular, multiple models for nonlinear control have a long tradition (Murray-Smith and Johansen, 1997).

In the data analysis world, hybridization in a broader sense is seen in efforts to combine two or more of Neural Network, Bayesian, GA, fuzzy logic and knowledge-based systems. The goal is to incorporate diverse sources and forms of information and to exploit the somewhat complementary nature of different methodologies. Since in real-life applications, classification is often not a stand-alone problem but rather a part of a larger system involving optimization, explanation and evaluation of decisions, interaction with the environment, etc.

2.4. FUSION OR MULTIPLE CLASSIFICATION APPROACHES

Fusion classifiers or multiple classifier systems (MCS) have received considerable attention in applied statistics (Hastie et al., 2001), machine learning (Dietterich, 2000) and pattern recognition (Kuncheva, 2004) for over a decade. Several studies demonstrate that the practice of combining several base classifier models into one aggregated classifier leads to significant gains in classification performance over its constituent members (Bauer and Kohavi, 1999).

Over the years, different fusion algorithms have been proposed, which differ along three structural dimensions of fusion design.

(i) The choice of the base or member classifier
(ii) The treatment of the input training data and
(iii) The aggregation strategy for the outputs of member classifiers.

Firstly, two broad strategies exist for choosing the members of a fusion classifier (Canuto et al., 2007). In hybrid multiple classifier, different types of algorithms are combined, whilst in non-hybrid multiple classifiers, one classifier algorithm is chosen as base classifier and replicated multiple times in order to constitute a fusion.

Secondly, many algorithms differ in terms of the treatment of the training data, used as input for each base classifier. Possibilities include data sampling schemes (Breiman, 1996), variable selection (Ho, 1998) or more complex data transformations (Kuncheva and Rodriguez, 2007; Rodriguez et al., 2006).

A third design characteristic involves the fusion rule used for the fusion of member outputs, ranging from simple average aggregation to more complex combination rules (Skurichina and Duin, 2000).

The most popular multiple classifier schemes are non-hybrid, where a base classification algorithm is applied to differently permutated training sets. A well-known method in this category is Bagging (Breiman, 1996), an acronym of bootstrap aggregating. Although numerous variations have been proposed since its introduction (Bauer and Kohavi, 1999; Bühlmann, 2002; Croux et al., 2007; Hothorn and Lausen, 2005),

Breiman's original implementation is still a widely used multiple classifier. In Bagging, each member is trained on a bootstrap sample of the training data, i.e. a random sample of observations drawn with replacement and having the same

size as the original training data. Multiple classifications is obtained by means of uniform majority voting, where an unlabeled observation is assigned the class with the highest number of votes among the individual classifiers' predictions. Theoretically, bootstrapping can induce large differences in the constructed individual classifiers which substantially improve the accuracy of the fusion classifier (Breiman, 1996).

Several variations upon Bagging have been proposed in search for further performance improvements. Two popular strategies involve

(i) Increasing variation in the training data for base classifiers and
(ii) The use of alternative base classifier algorithms.

Firstly, several studies have shown the impact of variations of the input data used for the training of base classifiers. Varying the training data of the members of an fusion classifier is a strategy to increase diversity amongst member classifiers, which is generally perceived as a key driver of fusion performance (Kuncheva and Whitaker, 2003).

In the Random Subspace Method (RSM), variables are randomly sampled to create training data sets for a decision tree classification fusion. RSM, also referred to as Attribute Bagging (Bryll et al., 2003), specifies that each fusion member is trained using a random feature subset (RFS), i.e. a random selection of explanatory variables sampled without replacement and of a predefined size.

A related method is the Random Forest algorithm by Breiman (2001), which has demonstrated high classification performance in many fields of research (e.g. Archer and Kirnes, 2008; Diaz-Uriate and de Andres, 2006; Gislason et al., 2006; Prasad et al., 2006; Svetnik et al., 2003). A Random Forest combines Bagging and a specific form of RSM where random feature subset selection is performed at each node of a member decision tree. More recently, Rodriguez et al.

(2006) proposed Rotation Forest, an multiple classifier based on rotations of the feature space through principal component analysis (PCA). The purpose of Rotation Forest is to increase the individual classifier performance and the diversity within the multiple classification process. Diversity is achieved for each classifier by applying feature extraction, while one tries to increase the performance by using all principal components and training the model on the whole data set.

A second strategy to increase classification performance is to select an alternative base classifier algorithm. Many studies have proposed fusion based on alternative base classifiers, such as Artificial Neural Networks (Hansen and Salamon, 1990; Maclin and Shavlik, 1995; Opitz and Shavlik, 1996; Schwenk and Bengio, 2000; Zhou et al., 2002), Support Vector Machines (Kim et al., 2002, 2003), parametric regression techniques (Prinzie and Van den Poel, 2008) and nonparametric regression techniques (Borra and Di Ciaccio, 2002).

2.5. ADVANCED CLASSIFICATION APPROACHES

In recent years, many advanced classification approaches, such as artificial Neural Networks, fuzzy-sets, and expert systems, have been widely applied for image classification. In general, image classification approaches can be grouped as

- Supervised and Unsupervised
- Parametric and Nonparametric
- Hard and Soft (fuzzy) classification
- Per-pixel, Sub-pixel, and Per-field.

A consolidation of the various works is given in Table 2.1.

2.6. EVALUATION OF CLASSIFICATION PERFORMANCE

Evaluation of classification results is an important process in the classification procedure. Different approaches may be employed, ranging from a qualitative evaluation based on expert knowledge to a quantitative accuracy assessment based on sampling strategies. To evaluate the performance of a classification method, Cihlar et al. (1998) proposed six criteria: accuracy, reproducibility, robustness, ability to fully use the information content of the data, uniform applicability, and objectiveness. In reality, no classification algorithm can satisfy all these requirements nor be applicable to all studies, due to different environmental settings and datasets used.

DeFries and Chan (2000) suggested the use of multiple criteria to evaluate the suitability of algorithms. These criteria include classification accuracy, computational resources, stability of the algorithm, and robustness to noise in the training data. Classification accuracy assessment is, however, the most common approach for an evaluation of classification performance.

Before implementing a classification accuracy assessment, the sources of errors have to be identified (Powell et al., 2004). In addition to errors from the classification itself, other sources of errors, such as position errors resulting from the registration, interpretation errors, and poor quality of training or test samples, all affect classification accuracy. In the process of accuracy assessment, it is commonly assumed that the difference between an image classification result and the reference data is due to the classification error.

Meanwhile, many authors (Janssen and van der Wel, 1994; Smits et al., 1999), have conducted reviews on classification accuracy assessment. They have assessed the status of accuracy assessment of image classification and discussed relevant issues.

TABLE 2.1

CONSOLIDATION OF SINGLE AND MULTIPLE CLASSIFICATION TECHNIQUES

Category	**Classifiers**
Per-pixel algorithms	Neural Network Decision Tree Spectral angle classifier Multistage classifier Enhancement-classification approach Multiple-Forward-Mode approach Radial Basis Function (RBF) and Markov Random Field (MRF) classifiers Classification by progressive generalization Support Vector Machine Unsupervised classification based on Independent Component Analysis Layered classification Nearest-Neighbor classification Selected pixel classification
Subpixel algorithms	Fuzzy Classifier Fuzzy Neural Network classification Rule-based machine learning classification
Per-field algorithms	Map-guided classification Object-oriented classification Graph-based, structural pattern recognition system
Contextual-based approaches	Frequency-based contextual classifier Fuzzy contextual classification Hierarchical classification

Knowledge-based algorithms	Knowledge-based classification Evidential reasoning classification Rule-based syntactical approach Visual fuzzy classification
Combinative approaches of multiple classifiers	Multiple classifier system Neural Network (MLC, Expert System) Neuro-fuzzy image classification Spectral and contextual classifiers Mixed contextual and per-pixel classification Neural Network and statistical classifiers Neural Network and Decision tree classification Combined supervised and unsupervised classification

A classification accuracy assessment generally includes three basic components: sampling design, response design, and estimation and analysis procedures (Stehman and Czaplewski, 1998). Selection of a suitable sampling strategy is a critical step (Congalton, 1991). The major components of a sampling strategy include sampling unit (pixels or polygons), sampling design, and sample size (Muller et al. 1998). Possible sampling designs include random, stratified random, systematic, double, and cluster sampling. A detailed description of sampling techniques can be found in Stehman and Czaplewski (1998) and Congalton and Green (1999).

The error matrix (confusion matrix) approach is the one most widely used in accuracy assessment (Foody, 2002). In order to properly generate a confusion matrix, the following factors must be considered (Congalton and Plourde, 2002)

(1) Reference data collection

(2) Classification scheme

(3) Sampling scheme

(4) Spatial autocorrelation and

(5) Sample size and sample unit

After generation of a confusion matrix, other important accuracy assessment elements, such as overall accuracy, true positive, true negatives and overall error rate can be derived. Congalton and Plourde (2002) and Foody(2002b, 2004a) have defined the meanings and provided computation methods for these elements.

Congalton and Green (1999) systematically reviewed the concept of basic accuracy assessment and some advanced topics involved in fuzzy-logic and multilayer assessments, and explained principles and practical considerations in designing and conducting accuracy assessment of remote-sensing data. The Kappa coefficient is a measure of overall statistical agreement of an error matrix, which takes non-diagonal elements into account.

The error matrix approach is only suitable for 'hard' classification, assuming that the map categories are mutually exclusive and exhaustive and that each location belongs to a single category. This assumption is often violated, especially for classifications with coarse spatial resolution imagery. 'Soft' classifications have been performed to minimize the mixed pixel problem using a fuzzy logic. The traditional confusion matrix approach is not appropriate for evaluating these soft classification results. Accordingly, many new measures, such as conditional entropy and mutual information (Finn, 1993, Maselli et al., 1994), fuzzy-set approaches (Gopal and Woodcock, 1994, Binaghi et al., 1999, Woodcock and Gopal 2000), symmetric index of information closeness (Foody, 1996), Renyi generalized entropy function (Ricotta and Avena, 2002), and

parametric generalization of Morisita's index (Ricotta, 2004) have been developed. In summary, the confusion matrix approach is the most common accuracy assessment approach for image classification approaches.

2.7. CONCLUSION

Although many classification approaches have been developed, which approach is suitable for a given application area is not fully understood. Selection of a suitable classifier requires consideration of many factors, such as classification accuracy, algorithm performance, and computational resources. As a general solution, a comparative study of different classifiers is often conducted to find the best classification result for a specific study (Keuchel et al. 2003; Erbek et al. 2004, South et al. 2004). While considering the usage of classifiers to classify handwritten characters, statistical classifiers, Support Vector Machines (SVM), Back Propagation Neural Networks (NN) and Multilayer Perceptron are more frequently used. Usage of multiple classifiers for classifying Tamil characters is still sparse. Multiple classification techniques are more popular with satellite or natural scene classification, where it has proved to be more efficient than the usage of single classifier.

Thus, in this research three classifiers, namely, BackPropagation Neural Network, Support Vector Machine and K Nearest Neighbor, are combined to form two-class and three-class multiple classifiers for classifying the various Tamil characters. The working of the individual classifiers and fusion methodology is described in the next chapter, Methodology.